

Quest® Change Auditor 7.3
SIEM Integration Guide



© 2022 Quest Software Inc.

ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.
Attn: LEGAL Dept.
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our website (<https://www.quest.com>) for regional and international office information.


Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at <https://www.quest.com/legal>.

Trademarks

Quest, the Quest logo, and Join the Innovation are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit <https://www.quest.com/legal/trademark-information.aspx>. All other trademarks and registered trademarks are property of their respective owners.

Legend

 **CAUTION:** A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

 **IMPORTANT NOTE, NOTE, TIP, MOBILE, or VIDEO:** An information icon indicates supporting information.

Contents

Integrating Change Auditor and SIEM Tools	5
Webhooks in Change Auditor	6
Webhook terminology	6
Subscription configuration process	7
Subscription Management	8
Adding the PowerShell module	9
Viewing available commands and help	9
Connecting to Change Auditor	9
Managing subscriptions	10
New-CAEventWebhookSubscription	10
Get-CAEventWebhookSubscriptions	11
Set-CAEventWebhookSubscription	13
Remove-CAEventWebhookSubscription	14
Get-CAEventExportSubsystems	14
Working with event subscriptions in the client	15
Managing a Splunk integration	15
Working with Splunk subscriptions through the client	16
New-CASplunkEventSubscription	17
Get-CASplunkEventSubscriptions	18
Set-CASplunkEventSubscription	19
Remove-CASplunkEventSubscription	21
Managing an IBM QRadar integration	21
Working with QRadar subscriptions through the client	22
New-CAQRadarExtension	24
New-CAQRadarEventSubscription	24
Get-CAQRadarEventSubscriptions	26
Set-CAQRadarEventSubscription	27
Remove-CAQRadarEventSubscription	28
Managing a Micro Focus Security ArcSight Logger and Enterprise Security Manager (ESM) integration	29
Working with Change Auditor data within ArcSight	29
Working with ArcSight subscriptions through the client	30
New-CAArcSightEventSubscription	31
Get-CAArcSightEventSubscriptions	32
Set-CAArcSightEventSubscription	33
Remove-CAArcSightEventSubscription	34
Managing a Quest IT Security Search integration (Preview)	35
Working with a Quest IT Security Search subscriptions through the client	35
New-CAITSSEventSubscription	36
Get-CAITSSEventSubscriptions	38
Set-CAITSSEventSubscription	39
Remove-CAITSSEventSubscription	40
Managing a Syslog integration	41

Working with Syslog subscriptions through the client	41
New-CASyslogEventSubscription	42
Get-CASyslogEventSubscriptions	43
Set-CASyslogEventSubscription	45
Remove-CASyslogEventSubscription	46
Managing a Microsoft Sentinel integration	47
Working with Microsoft Sentinel subscriptions through the client	47
New-CASentinelEventSubscription	48
Get-CASentinelEventSubscriptions	49
Set-CASentinelEventSubscription	50
Remove-CASentinelEventSubscription	51
Webhook technical insights	53
Handling webhook responses	53
About us	54
Our brand, our vision. Together.	54
Contacting Quest	54
Technical support resources	54

Integrating Change Auditor and SIEM Tools

- [Webhooks in Change Auditor](#)
- [Webhook terminology](#)
- [Subscription configuration process](#)

Webhooks in Change Auditor

Change Auditor administrators can configure Change Auditor to send events to a third party tool using webhook technology. This technology allows you to integrate Change Auditor with Splunk, IBM QRadar, Micro Focus Security ArcSight, Quest IT Security Search, or any other tool that accepts webhook notifications.

This guide is intended for customers who want to access and reuse the rich event data gathered by Change Auditor. It describes the configuration required to implement an integration with third-party tools.

Webhook terminology

- **Webhook receiver:** A service that has one or more webhook endpoints and is a third party tool that can receive Change Auditor events.
- **Webhook endpoint:** Specified with the NotificationUrl parameter, it is the web location where events are sent to inside the receiver.
- **Webhook subscription:** A configuration that contains information on the webhook receiver and how events should be sent. This includes a notification URL, notification interval, and the coordinator responsible for event forwarding.
- **Notification:** A message sent to the webhook receiver that contains a batch of events.

Subscription configuration process

To begin receiving event data, you need to:

- 1 Deploy a SIEM tool that can receive and process events from Change Auditor.
Create a webhook endpoint and configure it in your SIEM tool. The specifics for this are dependent on the tool that you are using.
Test the webhook receiver to confirm it is working properly.
- 2 Create and configure a subscription within Change Auditor. The subscription contains information such as where to send the events, which events to include, and the coordinator responsible for event forwarding. It also contains the heartbeat notification which is a message sent to the webhook receiver that notifies it that the Change Auditor coordinator is responsive. The heartbeat notification contains the bookmark time. The bookmark is the time the last event was sent in the event notification.
For details on creating subscriptions see [Managing a Splunk integration](#), [Managing an IBM QRadar integration](#), [Managing a Micro Focus Security ArcSight Logger and Enterprise Security Manager \(ESM\) integration](#), and [Managing a Quest IT Security Search integration \(Preview\)](#).
- 3 Once the subscription is created, the coordinator polls the database and continuously pushes new events to the specified notification URL in the subscription. Events are sent based on the time specified in the subscription.
- 4 Validate that events are being sent and processed by running the [Get-CAEventWebhookSubscriptions](#), [Get-CASplunkEventSubscriptions](#), [Get-CAQRadarEventSubscriptions](#), [Get-CAArcSightEventSubscriptions](#), or [Get-CAITSSEventSubscriptions](#) commands. The information in these commands indicate if the events are being received.

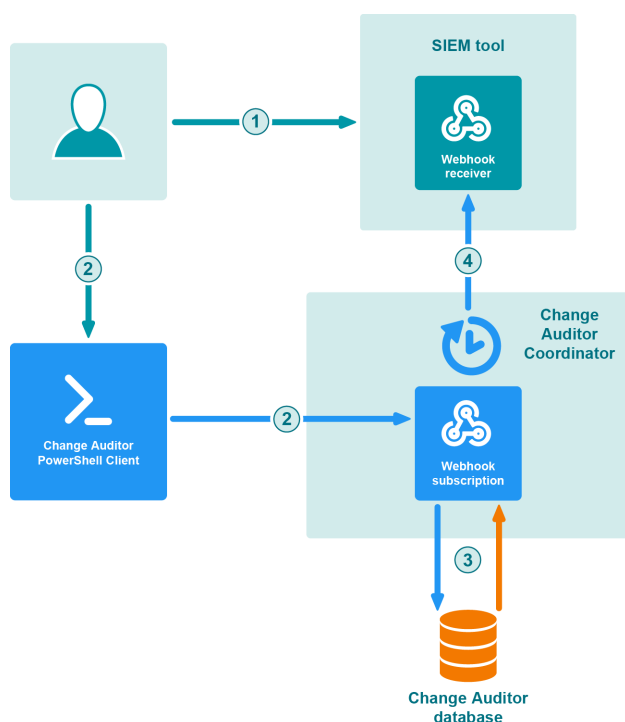


Figure 1. Webhook integration process

Subscription Management

- [Adding the PowerShell module](#)
- [Viewing available commands and help](#)
- [Connecting to Change Auditor](#)
- [Managing subscriptions](#)
- [Managing a Splunk integration](#)
- [Managing an IBM QRadar integration](#)
- [Managing a Micro Focus Security ArcSight Logger and Enterprise Security Manager \(ESM\) integration](#)
- [Managing a Quest IT Security Search integration \(Preview\)](#)
- [Managing a Syslog integration](#)
- [Managing a Microsoft Sentinel integration](#)

Adding the PowerShell module

Change Auditor comes with a PowerShell module for you to use to manage your environment. It is installed when you install the Windows client or a coordinator.

i | **NOTE:** Windows PowerShell version 3.0 or higher is required.

To import the Change Auditor PowerShell module:

- 1 Open a Windows PowerShell window and type the following at the Windows PowerShell command prompt:

```
Import-Module <path>
```

Where "<path>" is the file path for the ChangeAuditor.PowerShell.dll assembly found in the Change Auditor Windows client or Change Auditor coordinator folder.

- 2 To ensure that the module was added, type the following at the Windows PowerShell command prompt:

```
Get-Module -All
```

The registered PowerShell modules are listed.

Viewing available commands and help

- To view all available Change Auditor commands, enter:

```
Get-Command -Module ChangeAuditor.PowerShell
```

- To view help on each command including the syntax, enter:

```
Get-Help cmdletName
```

- To view an interactive command browser that shows you the layout of commands and the help for the commands, enter:

```
Show-Command cmdletName
```

i | **NOTE:** Sample scripts are available in the Change Auditor client folder. By default they are located here: C:\Program Files\Quest\ChangeAuditor\Client\PowerShell Sample Scripts

Connecting to Change Auditor

Most Change Auditor commands require a connection to a coordinator. You can make multiple connections to different coordinators or deployments in the same script as long as the version of Change Auditor is the same.

Example: Connect to the installation "XYZ" in the local forest

i | **NOTE:** This allows for fault tolerance if you have numerous coordinators by selecting the best option in the domain.

```
Connect-CAClient -InstallationName 'XYZ' -DomainName 'DomainName.com'
```

Managing subscriptions

To begin sending event data, you need to create a subscription with Change Auditor. The subscription contains information about the URL to send the notifications and heartbeats and the event subsystems to include.

- NOTE:** You must be a member of the Change Auditor Administrators group to run these commands.
- NOTE:** These are generic commands not tied to a specific SIEM tool.

- [New-CAEventWebhookSubscription](#)
- [Get-CAEventWebhookSubscriptions](#)
- [Set-CAEventWebhookSubscription](#)
- [Remove-CAEventWebhookSubscription](#)
- [Get-CAEventExportSubsystems](#)

New-CAEventWebhookSubscription

Use this command to create the subscription required to receive Change Auditor event data.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-NotificationUrl	Specifies where to send notifications. The notification URL is provided by the webhook receiver.
-Subsystems	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.
-StartTimeUTC (Optional)	Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created. For example: <ul style="list-style-type: none">• 20 July, 2020 12:01 PM uses local time• 2020-07-20 12:10:00Z uses UTC time The time will be local unless you specify the required flag to convert to UTC. NOTE: The time cannot be more than 30 days prior to the Change Auditor installation date.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. The URL is provided by the webhook receiver. NOTE: If no value is specified, heartbeat notifications are not sent.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the receiver. By default, this is set to 0, resulting in a continuous stream of events.

Parameter	Description
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to webhook receiver. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AuthorizationId (Optional)	Specifies the unique identifier used to confirm that the specified subscriber is authorized to accept event data. The Id is provided by webhook receiver.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to true.

Example: Create a subscription that sends O365, Active Directory, and Exchange events captured since March 1 to www.quest.com

```
$startTimeUTC = Get-Date "March 1, 2020 12:00 PM"
$notificationUrl = "https://www.quest.com/api/webhook"
$selectedSubsystems = Get-CAEventExportSubsystems -Connection $connection | Where-Object DisplayName -In -Value "Office 365", "Active Directory", "Exchange"
New-CAEventWebhookSubscription -Connection $connection -NotificationUrl $notificationUrl -StartTimeUTC $startTimeUTC -Subsystems $selectedSubsystems
```

Get-CAEventWebhookSubscriptions

Use this command to see the details of the current subscriptions.

Table 1. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SubscriptionId (optional)	The ID of an existing webhook subscription. If specified, the command will only return the webhook subscription with that ID. If not specified, all event subscriptions are returned. You can find the SubscriptionId by running this command using just the connection information. It is also returned by the New-CAEventWebhookSubscription command.

Example: List defined webhook subscriptions

```
Get-CAEventWebhookSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 2. Available configuration information

Setting	Description
Id	The subscription ID.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data being sent.
Enabled	Whether the subscription is enabled.
NotificationInterval	How often how often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
NotificationUrl	URL for event notifications.
HeartbeatUrl	URL for heartbeat notifications.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	The last event response. Provides the response in JSON format from the event receiver.
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	Last heartbeat response. (For example OK, HTTP 429 - Too many events being sent., and HTTP 401 - Unauthorized access.)
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.
BookmarkTime	Time of the event that was last sent.
AllowedCoordinators	List of coordinators permitted to send events.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator that sent events.
Internal	Whether this is an internal webhook created for a particular subscription.

Set-CAEventWebhookSubscription

Use this command to edit the subscription.

Table 3. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAEventWebhookStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified.
-NotificationUrl (Optional)	Specifies where to send notifications. The notification URL is provided by the webhook receiver.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. The URL is provided by the webhook receiver. NOTE: If no value is specified, heartbeat notifications are not sent.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the receiver. By default, this is set to 0, resulting in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to webhook receiver. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AuthorizationId (Optional)	Specifies the unique identifier used to confirm that the specified subscriber is authorized to accept event data. The Id is provided by webhook receiver.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems. NOTE: The subsystems specified override the current subsystems included in the subscription.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to true.

Example: Edit a webhook subscription to send events to www.quest.com for Office 365 and Active Directory

```
$subscriptionId = "ed01cc15-b67f-428d-b836-25405235dd1f"
```

```
$notificationUrl = "https://www.quest.com/api/webhook"
Set-CAEventWebhookSubscription -Connection $connection -SubscriptionId
$subscriptionId -NotificationUrl $notificationUrl
```

Example: Edit the subsystems included in a webhook subscription

```
$newSubsystems = Get-CAEventExportSubsystems -Connection $connection | ? {
$_.DisplayName -eq "File System" -or $_.DisplayName -eq "Active Directory" }
Set-CAEventWebhookSubscription -Connection $connection -SubscriptionId cd87b774-
8e65-46e1-8520-da478c60c4c3 -Subsystems $newSubsystems
```

Remove-CAEventWebhookSubscription

Use this command to remove a subscription.

NOTE: You cannot use this command to remove subscriptions that are marked as internal. You can use the `Get-CAEventWebhookSubscriptions` to see which subscriptions are internal.

Table 4. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAEventWebhookStatus object that corresponds to the subscription to remove. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the Subscription parameter is not specified. Use the Get-CAEventWebhookSubscriptions command to find the ID.

Example: Remove a webhook subscription

```
Remove-CAEventWebhookSubscription -Connection $connection -SubscriptionId
$subscriptionId
```

Get-CAEventExportSubsystems

Use this command to obtain an array of subsystems to include in a new subscriptions.

Table 5. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.

Example: Get the Active Directory and file system subsystems

```
Get-CAEventExportSubsystems -Connection $connection | ? {$_.DisplayName -eq "Active
Directory" -or $_.DisplayName -eq "File System" }
```

Working with event subscriptions in the client

The event subscriptions summary page displays the type of subscription (Target), where the events are being sent (Event URL), the subscription status (Enabled or Disabled), and when the last event was sent (Last Event).

From here, you can:

- View existing subscription details.
- Add ArcSight, Splunk, QRadar, IT Security Search, and Syslog subscriptions.
- Edit ArcSight, Splunk, QRadar, IT Security Search, and Syslog subscriptions.
- Create a QRadar extension.
- Remove a subscription.
- Enable and disable a subscription.

See [Managing a Splunk integration](#), [Managing an IBM QRadar integration](#), [Managing a Micro Focus Security ArcSight Logger and Enterprise Security Manager \(ESM\) integration](#), [Managing a Quest IT Security Search integration \(Preview\)](#), and [Managing a Syslog integration](#) for details.

Managing a Splunk integration

To begin to take advantage of the rich data gathered by Change Auditor by sending event data to Splunk, you need to create an event subscription with Change Auditor. The subscription contains information about where to send the notifications and heartbeats and the event subsystems to include.

i | **NOTE:** Columns that do not contain any event data will not display in Splunk.

i | **IMPORTANT:** To configure Splunk to receive events from Change Auditor you need to configure an HTTP event collector token in your Splunk instance.

- 1 Within Splunk, navigate to **Settings | Data Inputs | HTTP Event Collector**. Ensure that **All Tokens** are enabled under the Global Settings.
- 2 Click **New Token** and complete the steps in the wizard.
- 3 Copy the token. This value is required to create a Splunk subscription in Change Auditor.

Currently, you can create and manage a subscription for managed and unmanaged Splunk Cloud and Splunk Enterprise editions through the Change Auditor client or through PowerShell commands.

- [Working with Splunk subscriptions through the client](#)
- [New-CASplunkEventSubscription](#)
- [Get-CASplunkEventSubscriptions](#)
- [Set-CASplunkEventSubscription](#)
- [Remove-CASplunkEventSubscription](#)

Working with Splunk subscriptions through the client

To create a Splunk subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Add Splunk Subscription** to enter the required information.
- 3 Specify where to send the event data by entering the event URL.

For a Splunk Enterprise instance, use `https://[hostname]:[port]/services/collector/event`.

[hostname] is the hostname of your Splunk instance, [port] is the port defined in your Splunk instance's HTTP Event Collector token page (default is 8088).

For a Splunk Cloud instance, use:

"`https://input-[hostname]:[port]/services/collector/event`".

[hostname] is available in the address bar of an open Splunk Cloud instance and the default port is 8088.

- 4 Enter the event token.

Splunk uses this unique identifier to confirm that the specified event URL is authorized to accept event data. The token value is created during the Splunk instance configuration.

- 5 Click **Next** to select the events to forward based on subsystem and event date. Once the subscription is created the starting event date and time cannot be changed.
 - By default, events start sending after the subscription is created. To change when to begin sending events, click **Send events starting** and select the desired date and time. The time cannot be more than 30 days prior to the Change Auditor installation date.
 - Select the subsystems to include in the subscription.
- 6 Click **Finish**.

To view existing Splunk subscription details:

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Expand the required subscription.

The summary page displays the type of subscription (Target), where the events are being sent (Event URL), the subscription status (Enabled or Disabled), and when the last event was sent (Last Event).

To edit the Splunk subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Edit**.
- 3 If required, enter the new URL and click **Next**.
- 4 If required, add and remove the subsystems included in the subscription.
- 5 Click **Finish**.

To remove a Splunk subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Delete**.
- 3 Confirm the removal.

To enable and disable a subscription

- When viewing the summary information, select the status column and choose to enable or disable the subscription as required.

To refresh the summary information

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Refresh**.

New-CASplunkEventSubscription

Use this command to create the subscription required to send Change Auditor event data to Splunk.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SplunkUrl	<p>Specifies the address of your Splunk instance that will receive the event data.</p> <ul style="list-style-type: none">• For a Splunk Enterprise instance, use <code>https://[hostname]:[port]/services/collector/event</code> ([hostname] is the hostname of your Splunk instance, [port] is the port defined in your Splunk instance's HTTP Event Collector token page (default is 8088))• For a Splunk cloud instance, use: <code>"https://input-[hostname]:[port]/services/collector/event"</code>. (Hostname is available in the address bar of an open Splunk cloud instance.) <p>For details, see the Splunk documentation on HTTP Event Collector data inputs.</p>
-EventToken	<p>The unique identifier (token) used by Splunk to confirm that the specified Splunk URL is authorized to accept event data.</p> <p>The token value is created during the Splunk instance configuration.</p> <p>For details on creating an event collector token, see the Splunk documentation on HTTP Event Collector data inputs.</p>
-Subsystems	<p>Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems.</p> <p>NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.</p>
-StartTimeUTC (Optional)	<p>Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created.</p> <p>For example:</p> <ul style="list-style-type: none">• 20 July, 2020 12:01 PM uses local time• 2020-07-20 12:10:00Z uses UTC time <p>The time will be local unless you specify the required flag to convert to UTC.</p> <p>NOTE: The time cannot be more than 30 days prior to the Change Auditor installation date.</p>
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.

Parameter	Description
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. NOTE: If no value is specified, heartbeat notifications are not sent.
-HeartbeatToken (Optional)	The unique identifier (token) used by Splunk to confirm that the specified HeartbeatUrl is authorized to accept heartbeat notifications. NOTE: This is optional as you may have opted to send your heartbeat notifications to a URL that does not require a token for verification.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the Splunk instance. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatUrl. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to true.

Example: Create a subscription to send all subsystems event data to a Splunk instance

```
$allSubsystems = Get-CAEventExportSubsystems -Connection $connection
New-CASplunkEventSubscription -Connection $connection -SplunkUrl $splunkUrl -
EventToken $eventToken -Subsystems $allSubsystems
```

Get-CASplunkEventSubscriptions

Use this command to see the details of the current Splunk subscriptions.

i | **NOTE:** The “Batches sent”, “Last event time in UTC”, “Last event response” and “Events sent” are all indicators that the events are being received by Splunk. Any failures receiving the data populate the “Last event response” property in the object with information on why the data was not received.

Table 6. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SubscriptionId (optional)	The ID of an existing Splunk subscription. If specified, the command will only return the Splunk subscription with that ID. If not specified, all Splunk subscriptions are returned. You can find this by running this command using just the connection information. It is also returned by the New-CASplunkEventSubscription command.

Example: List defined Splunk subscriptions

```
Get-CASplunkEventSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 7. Available configuration information

Setting	Description
Id	The subscription ID.
WebhookSubscriptionId	The webhook subscription ID.
SplunkUrl	The URL where event data is sent.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data you want to send.
Enabled	Whether the subscription is enabled.
HeartbeatUrl	URL for heartbeat notifications.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	Last event response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	Last heartbeat response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.
NotificationInterval	How often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
AllowedCoordinators	List of coordinators permitted to send events.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator sending the events.
IncludeO365AADDDetails	Identifies whether or not the additionalDetails field with the raw JSON string is included for Office 365 and Azure Active Directory events.

Set-CASplunkEventSubscription

Use this command to modify a Splunk subscription.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAEventWebhookStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified. Use the Get-CASplunkEventSubscriptions command to get a list of objects.
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified. Use the Get-CASplunkEventSubscriptions command to find the ID.
-SplunkUrl (Optional)	<p>Specifies the address of your Splunk instance that will receive the event data.</p> <ul style="list-style-type: none"> For a Splunk Enterprise instance, use <code>https://[hostname]:[port]/services/collector/event</code> ([hostname] is the hostname of your Splunk instance, [port] is the port defined in your Splunk instance's HTTP Event Collector token page (default is 8088)) For a Splunk cloud instance, use: <code>"https://input-[hostname]:[port]/services/collector/event"</code>. (Hostname is available in the address bar of an open Splunk cloud instance.) <p>For details, see the Splunk documentation on HTTP Event Collector data inputs.</p>
-EventToken (Optional)	<p>The unique identifier (token) used by Splunk to confirm that the specified SplunkUri is authorized to accept event data.</p> <p>The token value is created during the Splunk instance configuration.</p> <p>For details on creating an event collector token, see the Splunk documentation on HTTP Event Collector data inputs.</p>
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	<p>Specifies where (URL) to send heartbeat notifications.</p> <p>NOTE: If no value is specified, heartbeat notifications are not sent.</p>
-HeartbeatToken (Optional)	<p>The unique identifier (token) used by Splunk to confirm that the specified heartbeatUri is authorized to accept heartbeat notifications.</p> <p>NOTE: This is optional as you may have opted to send your heartbeat notifications to a URL that does not require a token for verification.</p>
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the Splunk instance. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	<p>Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events.</p> <p>NOTE: The list order does not determine which coordinator is selected to send events.</p>

Parameter	Description
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the <code>Get-CAEventExportSubsystems</code> command and filter the list to specify the required subsystems. NOTE: The subsystems specified override the current subsystems included in the subscription.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named <code>additionalDetails</code> , containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the <code>additionalDetails</code> field is not included. By default, this is set to true.

Example: Disable a subscription

```
Set-CASplunkEventSubscription -Connection $connection -SubscriptionId $SubscriptionId -Enabled $false
```

Example: Edit the subsystems included in a webhook subscription

```
$newSubsystems = Get-CAEventExportSubsystems -Connection $connection | ? {
    $_.DisplayName -eq "File System" -or $_.DisplayName -eq "Active Directory" }
Set-CASplunkEventSubscription -Connection $connection -SubscriptionId cd87b774-8e65-46e1-8520-da478c60c4c3 -Subsystems $newSubsystems
```

Remove-CASplunkEventSubscription

Use this command to remove a Splunk subscription.

Table 8. Available parameters

Parameter	Description
-Connection	A connection obtained by using the <code>Connect-CAClient</code> command. See the Change Auditor Command Guide for details.
-Subscription	The <code>PSCAEventWebhookStatus</code> object that corresponds to the subscription to remove. This parameter is required if the <code>SubscriptionId</code> parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the <code>Subscription</code> parameter is not specified. Use the Get-CASplunkEventSubscriptions command to find the ID.

Example: Remove a Splunk subscription

```
Remove-CASplunkEventSubscription -Connection $connection -SubscriptionId $subscriptionId
```

Managing an IBM QRadar integration

You can take advantage of the rich data gathered by Change Auditor and use it with QRadar on-premises deployments. To begin sending event data, you need to create the QRadar extension and a QRadar event

subscription with Change Auditor. The subscription contains information about where to send the notifications and heartbeats and the event subsystems to include.

i **IMPORTANT:** To ensure that QRadar can read and present Change Auditor events, you need to import the extension created during the subscription creation or with the [New-CAQRadarExtension](#) command.

- 1 Open the QRadar console and select the **Admin** tab.
- 2 Select **Extensions Management | Add**.
- 3 Select the extension zip file and follow the instructions.

If prompted that the extension is not signed, select **Install**. When prompted to overwrite or keep existing data, select **Overwrite**.

i **IMPORTANT:** If your Change Auditor coordinator IP addresses change, you must update the corresponding log source identifier in QRadar.

- 1 Open the QRadar console and select the **Admin** tab.
- 2 Select **Log Sources**.
- 3 Select the Change Auditor log source and select **Edit**.
- 4 Enter the new IP address into the Log Source Identifier field and select **Save**.

- [Working with QRadar subscriptions through the client](#)
- [New-CAQRadarExtension](#)
- [New-CAQRadarEventSubscription](#)
- [Get-CAQRadarEventSubscriptions](#)
- [Set-CAQRadarEventSubscription](#)
- [Remove-CAQRadarEventSubscription](#)

Working with QRadar subscriptions through the client

i **NOTE:** All new subscriptions created through the client are encrypted with TLS/SSL. QRadar must be configured to accept the TLS protocol.

To create a subscription that is not encrypted, use the [New-CAQRadarEventSubscription](#) command and set `-TlsEnabled` to false.

To create a QRadar subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Add QRadar Subscription** to open the event subscription wizard.
- 3 Specify the IPv4 address or FQDN (fully qualified domain name) of the QRadar instance that will receive the event data.
- 4 Specify the port number for your QRadar instance that will receive the event data. The default port is 6514.
- 5 Click **Next** to select the events to forward based on subsystem and event date. Once the subscription is created the starting event date and time cannot be changed.
 - By default, events start sending after the subscription is created. To change when to begin sending events, click **Send events starting** and select the desired date and time. The time cannot be more than 30 days prior to the Change Auditor installation date.
 - Select the subsystems to include in the subscription.

- 6 Click **Next** to create the required extension to import to your QRadar instance. The extension instructs QRadar on how to read and present Change Auditor events. Specifically, it defines the log source (coordinator) and maps Change Auditor event columns to QRadar event columns.
 - i** | **NOTE:** If you create a new extension, the subscription is created in the disabled state. After you have imported and applied the extension, you need to enable the subscription.
 - i** | **NOTE:** If you have previously configured your QRadar instance for Change Auditor, you can select **My QRadar instance is already configured** and click **Finish** to complete the subscription setup.
- 7 Specify the file path and name for the file and click **Generate extension**.
- 8 Click **OK** in the confirmation dialog. Copy the file path to import the extension to your QRadar instance.
- 9 Click **Finish**.

To view existing QRadar subscription details:

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Expand the required subscription.

The summary page displays the type of subscription (Target), where the events are being sent, the subscription status (Enabled or Disabled), and when the last event was sent (Last Event).

To create a new extension

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Right-click the required subscription and click **Generate Extension**.
- 3 Specify the file path and name for the file and click **Generate file**.
- 4 Click **OK** in the confirmation dialog.
- 5 Copy the file path to import the extension to your QRadar instance.

To edit the QRadar subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Edit**.
- 3 If required, enter the server and port and click **Next**.
- 4 If required, add and remove the subsystems included in the subscription and click **Next**.
- 5 If required, you can generate a new extension.
- 6 Click **Finish**.

To remove a QRadar subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Delete**.
- 3 Confirm the removal.

To enable and disable a subscription

- When viewing the summary information, select the status column and choose to enable or disable the subscription as required.

To refresh the summary information

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Refresh**.

New-CAQRadarExtension

The Change Auditor extension must be added to QRadar for it to read and present Change Auditor events. Specifically, the extension defines the log source (coordinator) and maps Change Auditor event columns to QRadar event columns.

Use this command to create and generate a zip file that contains XML with the required extension. The extension must then be imported to QRadar.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SubscriptionId	The ID of an existing QRadar subscription. The subscription specifies the TLS log source port in the extension.
-ExtensionFilepath	Specifies the path for the output zip file.
-CoordinatorHosts (Optional)	Specifies a list of addresses from which QRadar can receive events.

Example: Create a QRadar subscription extension, and specify the location for the output and the TLS log source

```
New-CAQRadarExtension -Connection $connection -ExtensionFilepath $ExtensionFilepath -SubscriptionId $SubscriptionId
```

New-CAQRadarEventSubscription

Use this command to create the subscription required to send Change Auditor event data to QRadar.

i | **NOTE:** Some Change Auditor events exceed QRadar's recommended supported event data length. If a Change Auditor event exceeds this limit, the event data is continued in new QRadar events to ensure all data is stored.

i | **NOTE:** Columns that do not contain any event data will not display in QRadar.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-QRadarHost	Specifies the IPv4 address or FQDN (fully qualified domain name) of your QRadar instance that will receive the event data.
-Subsystems	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.
-QRadarPort (Optional)	Specifies the port number for your QRadar instance that will receive the event data. When TlsEnabled is set to true to enable TLS/SSL encryption, the default port is 6514. When the TlsEnabled is set to false to send unencrypted events, the default port is 514.

Parameter	Description
-StartTimeUTC (Optional)	<p>Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created.</p> <p>For example:</p> <ul style="list-style-type: none"> • 20 July, 2020 12:01 PM uses local time • 2020-07-20 12:10:00Z uses UTC time <p>The time will be local unless you specify the required flag to convert to UTC.</p> <p>NOTE: The time cannot be more than 30 days prior to the Change Auditor installation date.</p>
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	<p>Specifies where (URL) to send heartbeat notifications. Heartbeat notifications cannot be sent directly to QRadar. To use this parameter, you must use a previously created webhook URL.</p> <p>NOTE: If no value is specified, a heartbeat notification is not sent.</p>
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the QRadar instance. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	<p>Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events.</p> <p>NOTE: The list order does not determine which coordinator is selected to send events.</p>
-TlsEnabled (Optional)	<p>When set to true, the subscription sends events encrypted with TLS/SSL and sets the default port to 6514.</p> <p>When set to false, the subscription sends events without encryption enabled, and sets the default port to 514.</p> <p>The default is set to true.</p>
-IncludeO365AADDetails (Optional)	<p>Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included.</p> <p>By default, this is set to false.</p>

Example: Create a subscription to send all subsystems event data to a QRadar instance

```
$allSubsystems = Get-CAEventExportSubsystems -Connection $connection
New-CAQRadarEventSubscription -Connection $connection -QRadarHost $QRadarHost
-Subsystems $allSubsystems
```

Get-CAQRadarEventSubscriptions

Use this command to see the details of the current QRadar subscriptions.

i | **NOTE:** The “Batches sent”, “Last event time in UTC”, “Last event response” and “Events sent” are all indicators that the events are being received by QRadar. Any failures receiving the data populate the “Last event response” property in the object with information on why the data was not received.

Table 9. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SubscriptionId (optional)	The ID of an existing QRadar subscription. If specified, the command will only return the QRadar subscription with that ID. If not specified, all QRadar subscriptions are returned.

Example: List defined QRadar subscriptions

```
Get-CAQRadarEventSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 10. Available configuration information

Setting	Description
Id	The subscription ID.
WebhookSubscriptionId	The webhook subscription ID.
QRadarUrl	The URL where the event data is sent.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data being sent.
Enabled	Whether the subscription is enabled.
HeartbeatUrl	The URL where heartbeat notifications are sent.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	Last event response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	Last heartbeat response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.
NotificationInterval	How often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.

Table 10. Available configuration information

Setting	Description
AllowedCoordinators	List of coordinators permitted to send events.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator sending the events.
IncludeO365AADDetails	Identifies whether or not the additionalDetails field with the raw JSON string is included for Office 365 and Azure Active Directory events.

Set-CAQRadarEventSubscription

i | **NOTE:** A new extension must be created if the -QRadarHost or -QRadarPort has been changed.

Use this command to modify a QRadar subscription.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAQRadarSubscriptionStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified. Use the Get-CAQRadarEventSubscriptions command to find the ID.
-QRadarHost	Specifies the IPv4 address or FQDN (fully qualified domain name) of your QRadar instance that will receive the event data.
-QRadarPort (Optional)	Specifies the port number for your QRadar instance that will receive the event data. The default port depends on whether the existing subscription is sending encrypted or unencrypted events. When TlsEnabled is set to true the default port is 6514; when set to false the default port is 514.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. NOTE: If no value is specified, a heartbeat notification is not sent.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the QRadar instance. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatUrl. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.

Parameter	Description
-Subsystems (Optional)	<p>Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems.</p> <p>NOTE: To obtain an array of subsystems, use the <code>Get-CAEventExportSubsystems</code> command and filter the list to specify the required subsystems.</p> <p>NOTE: The subsystems specified override the current subsystems included in the subscription.</p>
-IncludeO365AADDetails (Optional)	<p>Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named <code>additionalDetails</code>, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the <code>additionalDetails</code> field is not included.</p> <p>By default, this is set to false.</p>

Example: Disable a subscription

```
Set-CAQRadarEventSubscription -Connection $connection -QRadarHost $QRadarHost -
SubscriptionId $SubscriptionId -Enabled $false
```

Example: Edit the subsystems included in a webhook subscription

```
$newSubsystems = Get-CAEventExportSubsystems -Connection $connection | ? {
$_.DisplayName -eq "File System" -or $_.DisplayName -eq "Active Directory" }
Set-CAQRadarEventSubscription -Connection $connection -QRadarHost $QRadarHost -
SubscriptionId cd87b774-8e65-46e1-8520-da478c60c4c3 -Subsystems $newSubsystems
```

Remove-CAQRadarEventSubscription

Use this command to remove a QRadar subscription.

Table 11. Available parameters

Parameter	Description
-Connection	A connection obtained by using the <code>Connect-CAClient</code> command. See the Change Auditor Command Guide for details.
-Subscription	The <code>PSCAQRadarSubscriptionStatus</code> object that corresponds to the subscription to remove. This parameter is required if the <code>SubscriptionId</code> parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the <code>Subscription</code> parameter is not specified. Use the Get-CAQRadarEventSubscriptions command to find the ID.

Example: Remove a QRadar subscription

```
Remove-CAQRadarEventSubscription -Connection $connection -SubscriptionId
$subscriptionId
```

Managing a Micro Focus Security ArcSight Logger and Enterprise Security Manager (ESM) integration

You can take advantage of the rich data gathered by Change Auditor and use it with ArcSight Logger and ArcSight Enterprise Security Manager (ESM). To begin sending event data, you need to create an ArcSight event subscription with Change Auditor.

To send encrypted Change Auditor events to ArcSight ESM or ArcSight Logger, you must set the ArcSight host and port to match the host and port of the ArcSight connector configured to receive syslog messages over TCP.

When sending encrypted events, communication between the coordinator and connector is unencrypted, however, communication between the connector and ArcSight is encrypted. For improved security:

- Install the connector on the same computer as the coordinator sending events.
- Ensure that the coordinator is the only coordinator permitted to send events to the connector using the AllowedCoordinators parameter.

The subscription contains information about where to send the notifications and heartbeats and the event subsystems to include.

- [Working with Change Auditor data within ArcSight](#)
- [Working with ArcSight subscriptions through the client](#)
- [New-CAArcSightEventSubscription](#)
- [Get-CAArcSightEventSubscriptions](#)
- [Set-CAArcSightEventSubscription](#)
- [Remove-CAArcSightEventSubscription](#)

Working with Change Auditor data within ArcSight

The following table describes how Change Auditor event details are mapped to the event details provided in ArcSight's Common Event Format (CEF) extensions. All other Change Auditor columns not listed here will display as custom columns in ArcSight.

NOTE: Columns that do not contain any event data will not display in ArcSight.

Table 12. Mapping information

Change Auditor column	ArcSight column
Subsystem	deviceEventClassId
Event	name
Severity	agentSeverity
Action	categoryBehaviour
Result	categoryOutcome
Server FQDN	deviceHostName
IP Address	deviceAddress
ID	eventId
Origin IPv4	sourceAddress

Table 12. Mapping information

Change Auditor column	ArcSight column
Origin IPv6	c6a2
Origin	sourceHostName
User SID	sourceUserId
User	sourceUserName
Description	message
Time Detected	endTime
Time Detected	startTime

NOTE: Some Change Auditor events exceed ArcSight's recommended supported event data length. If a Change Auditor event exceeds this limit, the event data is continued in new ArcSight events to ensure all data is stored.

Working with ArcSight subscriptions through the client

To create an ArcSight subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Add ArcSight Subscription** to open the event subscription wizard.
- 3 Specify the IPv4 address or FQDN (fully qualified domain name) of the computer where ArcSight Logger or the ArcSight connector is installed.
- 4 Specify the port number for the ArcSight Logger or the ArcSight connector. The default port is 515.
- 5 Click **Next** to select the events to forward based on subsystem and event date. Once the subscription is created the starting event date and time cannot be changed.
 - By default, events start sending after the subscription is created. To change when to begin sending events, click **Send events starting** and select the desired date and time. The time cannot be more than 30 days prior to the Change Auditor installation date.
 - Select the subsystems to include in the subscription.
- 6 Click **Finish**.

To view existing ArcSight subscription details:

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Expand the required subscription.

The summary page displays the type of subscription (Target), where the events are being sent, the subscription status (Enabled or Disabled), and when the last event was sent (Last Event).

To edit the ArcSight subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Edit**.
- 3 If required, enter the server and port and click **Next**.
- 4 If required, add and remove the subsystems included in the subscription.
- 5 Click **Finish**.

To remove a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Delete**.
- 3 Confirm the removal.

To enable and disable a subscription

- When viewing the summary information, select the status column and choose to enable or disable the subscription as required.

To refresh the summary information

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Refresh**.

New-CAArcSightEventSubscription

Use this command to create the subscription required to send Change Auditor event data to ArcSight.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-ArcSightHost	Specifies the IP address or host name of the computer where ArcSight Logger or the ArcSight connector is installed.
-ArcSightPort (Optional)	The port number for the ArcSight Logger or the ArcSight connector. The default port is 515.
-Subsystems	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.
-StartTimeUTC (Optional)	Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created. For example: <ul style="list-style-type: none">• 20 July, 2020 12:01 PM uses local time• 2020-07-20 12:10:00Z uses UTC time The time will be local unless you specify the required flag to convert to UTC. NOTE: The time cannot be more than 30 days prior to the Change Auditor installation date.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. Heartbeat notifications cannot be sent directly to ArcSight. To use this parameter, you must use a previously created webhook URL. NOTE: If no value is specified, a heartbeat notification is not sent.

Parameter	Description
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the computer where ArcSight Logger or the ArcSight connector is installed. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat message.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to false.

Example: Create a subscription to send all subsystems event data to a computer where ArcSight Logger or the ArcSight connector is installed

```
$allSubsystems = Get-CAEventExportSubsystems -Connection $connection
New-CAArcSightEventSubscription -Connection $connection -ArcSightHost $ArcSightHost
-Subsystems $allSubsystems
```

Get-CAArcSightEventSubscriptions

Use this command to see the details of the current ArcSight subscriptions.

- NOTE:** The “Batches sent”, “Last event time in UTC”, “Last event response” and “Events sent” are all indicators that the events are being received by ArcSight. Any failures receiving the data populate the “Last event response” property in the object with information on why the data was not received.

Table 13. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SubscriptionId (optional)	The ID of an existing ArcSight subscription. If specified, the command will only return the ArcSight subscription with that ID. If not specified, all ArcSight subscriptions are returned.

Example: List defined ArcSight subscriptions

```
Get-CAArcSightEventSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 14. Available configuration information

Setting	Description
Id	The subscription ID.
WebhookSubscriptionId	The webhook subscription ID.
ArcSightHost	The IP address or host name where event data is sent.

Table 14. Available configuration information

Setting	Description
ArcSightPort	The port where event data is sent.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data being sent.
Enabled	Whether the subscription is enabled.
HeartbeatUrl	URL for heartbeat notifications.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	Last event response.
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	The last heartbeat response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.
NotificationInterval	How often how often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
AllowedCoordinators	List of coordinators permitted to send events.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator sending the events.
IncludeO365AADDDetails	Identifies whether or not the additionalDetails field with the raw JSON string is included for Office 365 and Azure Active Directory events.

Set-CAArcSightEventSubscription

Use this command to modify an ArcSight subscription.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAArcSightEventSubscriptionStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified. Use the Get-CAArcSightEventSubscriptions command to find the ID.
-ArcSightHost (Optional)	Specifies the IP address or host name of the computer where ArcSight Logger or the ArcSight connector is installed.
-ArcSightPort (Optional)	The the port number for the ArcSight Logger or the ArcSight connector. The default port is 515.

Parameter	Description
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. NOTE: If no value is specified, a heartbeat notification is not sent.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the computer where ArcSight Logger or the ArcSight connector is installed. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatUrl. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems. NOTE: The subsystems specified override the current subsystems included in the subscription.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to false.

Example: Disable a subscription

```
Set-CAArcSightEventSubscription -Connection $connection -SubscriptionId $SubscriptionId -Enabled $false
```

Example: Edit the subsystems included in a webhook subscription

```
$newSubsystems = Get-CAEventExportSubsystems -Connection $connection | ? {
    $_.DisplayName -eq "File System" -or $_.DisplayName -eq "Active Directory" }
Set-CAArcSightEventSubscription -Connection $connection -SubscriptionId cd87b774-8e65-46e1-8520-da478c60c4c3 -Subsystems $newSubsystems
```

Remove-CAArcSightEventSubscription

Use this command to remove a subscription.

Table 15. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAArcSightEventSubscriptionStatus object that corresponds to the subscription to remove. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the Subscription parameter is not specified. Use the Get-CAArcSightEventSubscriptions command to find the ID.

Example: Remove an ArcSight subscription

```
Remove-CAArcSightEventSubscription -Connection $connection -SubscriptionId $subscriptionId
```

Managing a Quest IT Security Search integration (Preview)

Quest IT Security Search is a Google-like, IT search engine that enables IT administrators and security teams to quickly respond to security incidents and analyze event forensics.

To send the rich event gathered by Change Auditor to IT Security Search, you need to create an event subscription with Change Auditor. The subscription contains information about where to send the notifications and heartbeats and the event subsystems to include.

i | **NOTE:** The IT Security Search warehouse connector must be configured to receive Change Auditor events. See the IT Security Search Release Notes for details.

i | **NOTE:** This feature is supported as of IT Security Search version 11.4.1.

- [Working with a Quest IT Security Search subscriptions through the client](#)
- [New-CAITSSEventSubscription](#)
- [Get-CAITSSEventSubscriptions](#)
- [Set-CAITSSEventSubscription](#)
- [Remove-CAITSSEventSubscription](#)

i | **NOTE:** These commands are in preview mode for this release.

Working with a Quest IT Security Search subscriptions through the client

To create a IT Security Search subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Add IT Security Search Subscription** to enter the required information.
- 3 Specify where to send the event data by entering the event URL.
Use `https://[hostname]:[port]/warehouse/changeauditor/events`.

- 4 Enter the username, password, and domain used to sign into the IT Security Search client.
- 5 Click **Next** to select the events to forward based on subsystem and event date. Once the subscription is created the starting event date and time cannot be changed.
 - By default, events start sending after the subscription is created. To change when to begin sending events, click **Send events starting** and select the desired date and time. The time cannot be more than 30 days prior to the Change Auditor installation date.
 - Select the subsystems to include in the subscription.
- 6 Click **Finish**.

To view existing IT Security Search subscription details:

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Expand the required subscription.

The summary page displays the type of subscription (Target), where the events are being sent (Event URL), the subscription status (Enabled or Disabled), and when the last event was sent (Last Event).

To edit the IT Security Search subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Edit**.
- 3 If required, enter the new URL and click **Next**.
- 4 If required, add and remove the subsystems included in the subscription.
- 5 Click **Finish**.

To remove a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Delete**.
- 3 Confirm the removal.

To enable and disable a subscription

- When viewing the summary information, select the status column and choose to enable or disable the subscription as required.

To refresh the summary information

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Refresh**.

New-CAITSSEventSubscription

Use this command to create the subscription required to send Change Auditor event data to IT Security Search.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-ITSSUrl	Specifies the address of your IT Security Search instance that will receive the event data. <ul style="list-style-type: none"> By default, this is set to <code>http://[hostname]:[port]/warehouse/changeauditor/events</code>. Hostname is the IT Security Search instance and the default port is 8087.
-Subsystems	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. <p>NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.</p>
-Credential	Domain\Username and password used for authenticating with the IT Security Search server. Enter the username and password used to sign into the IT Security Search client.
-StartTimeUTC (Optional)	Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created. For example: <ul style="list-style-type: none"> 20 July, 2020 12:01 PM uses local time 2020-07-20 12:10:00Z uses UTC time The time will be local unless you specify the required flag to convert to UTC. <p>NOTE: The time cannot be more than 30 days prior to the Change Auditor installation date.</p>
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. <p>NOTE: If no value is specified, heartbeat notifications are not sent.</p>
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the IT Security Search instance. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatUrl. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. <p>NOTE: The list order does not determine which coordinator is selected to send events.</p>
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named <code>additionalDetails</code> , containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the <code>additionalDetails</code> field is not included. By default, this is set to true.

Example: Create a subscription to send all subsystems event data to an IT Search instance

```
$allSubsystems = Get-CAEventExportSubsystems -Connection $connection  
New-CAITSSSEventSubscription -Connection $connection -ITSSUrl $ITSSUrl -Credential  
$Credential -Subsystems $allSubsystems
```

Get-CAITSSSEventSubscriptions

Use this command to see the details of the current IT Security Search subscriptions.

- i** | **NOTE:** The “Batches sent”, “Last event time in UTC”, “Last event response” and “Events sent” are all indicators that the events are being received by IT Security Search. Any failures receiving the data populate the “Last event response” property in the object with information on why the data was not received.

Table 16. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-SubscriptionId (optional)	The ID of an existing IT Security Search subscription. If specified, the command will only return the subscription with that ID. If not specified, all IT Security Search subscriptions are returned. You can find this by running this command using just the connection information. It is also returned by the New-CAITSSSEventSubscription command.

Example: List defined IT Security Search subscriptions

```
Get-CAITSSSEventSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 17. Available configuration information

Setting	Description
Id	The subscription ID.
WebhookSubscriptionId	The webhook subscription ID.
ITSSUrl	The URL where event data is sent.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data you want to send.
Enabled	Whether the subscription is enabled.
HeartbeatUrl	URL for heartbeat notifications.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	Last event response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	Last heartbeat response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.

Table 17. Available configuration information

Setting	Description
NotificationInterval	How often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
AllowedCoordinators	List of coordinators permitted to send events.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator sending the events.
IncludeO365AADDetails	Identifies whether or not the additionalDetails field with the raw JSON string is included for Office 365 and Azure Active Directory events.

Set-CAITSSEventSubscription

Use this command to modify an IT Security Search subscription.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAITSSSubscriptionStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified. Use the Get-CAITSSEventSubscriptions command to get a list of objects.
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified. Use the Get-CAITSSEventSubscriptions command to find the ID.
-ITSSUrl	Specifies the address of your IT Security Search instance that will receive the event data. <ul style="list-style-type: none"> By default, this is set to <code>http://[hostname]:[port]/warehouse/changeauditor/events</code>. Hostname is the IT Security Search instance and the default port is 8087.
-Credential (Optional)	Domain\Username and password used for authenticating with the IT Security Search server. Enter the username and password used to sign into the IT Security Search client.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. NOTE: If no value is specified, heartbeat notifications are not sent.

Parameter	Description
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the IT Security Search instance. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems. NOTE: The subsystems specified override the current subsystems included in the subscription.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named <code>additionalDetails</code> , containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the <code>additionalDetails</code> field is not included. By default, this is set to true.

Example: Disable a subscription

```
Set-CAITSSEventSubscription -Connection $connection -ITSSUrl $ITSSUrl -
SubscriptionId $SubscriptionId -Enabled $false
```

Example: Edit the subsystems included in an IT Security Search subscription

```
$newSubsystems = Get-CAEventExportSubsystems -Connection $connection | ? {
$_.DisplayName -eq "File System" -or $_.DisplayName -eq "Active Directory" }
Set-CAITSSEventSubscription -Connection $connection -ITSSUrl $ITSSUrl -
SubscriptionId cd87b774-8e65-46e1-8520-da478c60c4c3 -Subsystems $newSubsystems
```

Remove-CAITSSEventSubscription

Use this command to remove an IT Security Search subscription.

Table 18. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCAITSSSubscriptionStatus object that corresponds to the subscription to remove. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the Subscription parameter is not specified. Use the Get-CAITSSEventSubscriptions command to find the ID.

Example: Remove an IT Security Search subscription

```
Remove-CAITSSEventSubscription -Connection $connection -SubscriptionId $subscriptionId
```

Managing a Syslog integration

To send the rich events gathered by Change Auditor to a Syslog server, you need to create an event subscription with Change Auditor. The subscription contains information about where to send the notifications and heartbeats and the event subsystems to include.

- [Working with Syslog subscriptions through the client](#)
- [New-CASyslogEventSubscription](#)
- [Get-CASyslogEventSubscriptions](#)
- [Set-CASyslogEventSubscription](#)
- [Remove-CASyslogEventSubscription](#)

Working with Syslog subscriptions through the client

To create a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Add Syslog Subscription** to enter the required information.
- 3 Specify the IPv4 address or FQDN (fully qualified domain name) of the computer where the Syslog server is installed.
- 4 Specify the port number.
- 5 Specify the message format (LEEF or CEF) and whether TLS encryption is required. By default, this is enabled.
- 6 Click **Next** to select the events to forward based on subsystem and event date. Once the subscription is created the starting event date and time cannot be changed.
 - By default, events start sending after the subscription is created. To change when to begin sending events, click **Send events starting** and select the desired date and time. The time cannot be more than 30 days prior to the Change Auditor installation date.
 - Select the subsystems to include in the subscription.
- 7 Click **Finish**.

To view existing subscription details:

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Expand the required subscription.

The summary page displays the type of subscription (Target), where the events are being sent (Destination), the subscription status (Enabled or Disabled), and when the last event was sent (Last Event).

To edit a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Edit**.
- 3 If required, enter the server and port.
- 4 If required, select the message format (LEEF or CEF) and whether TLS encryption is required.
- 5 Click **Next**.
- 6 If required, add and remove the subsystems included in the subscription.
- 7 Click **Finish**.

To remove a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Delete**.
- 3 Confirm the removal.

To enable and disable a subscription

- When viewing the summary information, select the status column and choose to enable or disable the subscription as required.

To refresh the summary information

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Refresh**.

New-CASyslogEventSubscription

Use this command to create the subscription required to send Change Auditor event data to a Syslog server.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor PowerShell Command Guide for details.
-Host (Optional)	Specifies the IP address or host name of the computer where the Syslog server is installed.
-Port (Optional)	Specifies the port of the computer where the Syslog server is installed.
-TlsEnabled (Optional)	Specifies whether TLS encryption is enabled.
-SyslogFormat (Optional)	Specifies the message format (LEEF or CEF).

Parameter	Description
-Subsystems	<p>Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems.</p> <p>NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.</p>
-StartTimeUTC (Optional)	<p>Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created.</p> <p>For example:</p> <ul style="list-style-type: none"> 20 July, 2020 12:01 PM uses local time 2020-07-20 12:10:00Z uses UTC time <p>The time will be local unless you specify the required flag to convert to UTC.</p> <p>NOTE: The time cannot be more than 30 days prior to the Change Auditor installation date.</p>
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	<p>Specifies where (URL) to send heartbeat notifications.</p> <p>NOTE: If no value is specified, a heartbeat notification is not sent.</p>
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the computer where the Syslog server is installed. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatURL. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat message.
-AllowedCoordinators (Optional)	<p>Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events.</p> <p>NOTE: The list order does not determine which coordinator is selected to send events.</p>
-IncludeO365AADDetails (Optional)	<p>Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named <code>additionalDetails</code>, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the <code>additionalDetails</code> field is not included.</p> <p>By default, this is set to true.</p>

Example: Create a subscription to send all subsystems event data to a computer where a Syslog server is installed

```
$allSubsystems = Get-CAEventExportSubsystems -Connection $connection
New-CASyslogEventSubscription -Connection $connection -Host $Host -Subsystems
$allSubsystems
```

Get-CASyslogEventSubscriptions

Use this command to see the details of the current Syslog subscriptions.

i **NOTE:** The “Batches sent”, “Last event time in UTC”, “Last event response” and “Events sent” are all indicators that the events are being received by a Syslog server. Any failures receiving the data populate the “Last event response” property in the object with information on why the data was not received.

Table 19. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor PowerShell Command Guide for details.
-SubscriptionId (optional)	The ID of an existing Syslog subscription. If specified, the command will only return the Syslog subscription with that ID. If not specified, all Syslog subscriptions are returned.

Example: List defined Syslog subscriptions

```
Get-CASyslogEventSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 20. Available configuration information

Setting	Description
Id	The subscription ID.
WebhookSubscriptionId	The webhook subscription ID.
Host	The IP address or host name where event data is sent.
Port	The port where event data is sent.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data being sent.
Enabled	Whether the subscription is enabled.
HeartbeatUrl	URL for heartbeat notifications.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	Last event response.
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	The last heartbeat response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.
NotificationInterval	How often how often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
AllowedCoordinators	List of coordinators permitted to send events.
LastCoordinator	The coordinator that is sending events. If the subscription is disabled, this is the last coordinator sending the events.
IncludeO365AADDetails	Identifies whether or not the additionalDetails field with the raw JSON string is included for Office 365 and Azure Active Directory events.
TlsEnabled	Specifies whether TLS encryption is enabled.
SyslogFormat	Specifies the message format (LEEF or CEF).

Set-CASyslogEventSubscription

Use this command to modify a Syslog subscription.

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Host (Optional)	Specifies the IP address or host name of the computer where the Syslog server is installed.
-Port (Optional)	The the port number for the Syslog server.
-TlsEnabled (Optional)	Specifies whether TLS encryption is enabled.
-SyslogFormat (Optional)	Specifies the message format (LEEF or CEF).
-Subscription	The PSCASyslogEventSubscriptionStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified. Use the Get-CASyslogEventSubscriptions command to find the ID.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 10000 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-HeartbeatUrl (Optional)	Specifies where (URL) to send heartbeat notifications. NOTE: If no value is specified, a heartbeat notification is not sent.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to the computer where the Syslog server is installed. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the HeartbeatUrl. By default, this is set to every 5 minutes. Setting this to 0 disables the heartbeat notifications.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems. NOTE: The subsystems specified override the current subsystems included in the subscription.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include a field named additionalDetails, containing the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to true.

Example: Disable a subscription

```
Set-CASyslogEventSubscription -Connection $connection -SubscriptionId
$SubscriptionId -Enabled $false
```

Example: Edit the subsystems included in a subscription

```
$newSubsystems = Get-CAEventExportSubsystems -Connection $connection | ? {
$_.DisplayName -eq "File System" -or $_.DisplayName -eq "Active Directory" }
Set-CASyslogEventSubscription -Connection $connection -SubscriptionId cd87b774-8e65-
46e1-8520-da478c60c4c3 -Subsystems $newSubsystems
```

Remove-CASyslogEventSubscription

Use this command to remove a subscription.

Table 21. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCASyslogEventSubscriptionStatus object that corresponds to the subscription to remove. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the Subscription parameter is not specified. Use the Get-CASyslogEventSubscriptions command to find the ID.

Example: Remove a subscription

```
Remove-CASyslogEventSubscription -Connection $connection -SubscriptionId
$subscriptionId
```

Managing a Microsoft Sentinel integration

To send the rich events gathered by Change Auditor to Microsoft Sentinel, you need to create an event subscription with Change Auditor. The subscription contains information about where to send the notifications and heartbeats and the event subsystems to include.

i **NOTE:** When enabled, heartbeat notifications are sent to the Heartbeat_ChangeAuditor_CL log in Microsoft Sentinel where you can query and alert on them if required.

The following Azure Log Analytics KQL query example, returns a list of servers missing heartbeats in the last 10 minutes where the servers have been active at some point in the last 24 hours.

```
Heartbeat_ChangeAuditor_CL
| where timeSent_t >= ago(24h)
| project timeSent_t, sender_s
| order by timeSent_t desc
| partition by sender_s ( top 1 by timeSent_t )
| where timeSent_t < ago(10m)
| project sender_s
```

- [Working with Microsoft Sentinel subscriptions through the client](#)
- [New-CASentinelEventSubscription](#)
- [Get-CASentinelEventSubscriptions](#)
- [Set-CASentinelEventSubscription](#)
- [Remove-CASentinelEventSubscription](#)

Working with Microsoft Sentinel subscriptions through the client

To create a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click **Add** then select **Microsoft Sentinel subscription** to enter the required information.
- 3 Specify the Workspace ID for the Log Analytics workspace that has been enabled for Microsoft Sentinel.
- 4 Specify the secret key for the Log Analytics workspace that has been enabled for Microsoft Sentinel.

i **NOTE:** The Workspace ID and the secret keys (Primary and Secondary) can be found in Settings on the Agents management page for the Log Analytics workspace configured for Microsoft Sentinel (Azure Portal).

- 5 Click **Next** to select the events to forward based on subsystem and event date. Once the subscription is created the starting event date and time cannot be changed.
 - By default, events start sending after the subscription is created. To change when to begin sending events, click **Send events starting from** and select the desired date and time.
 - Select the subsystems to include in the subscription.
- 6 Click **Finish**.

To view existing subscription details:

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.

- 2 Expand the required subscription.

The summary page displays the type of subscription (Target), where the events are being sent (Destination), the subscription status (Enabled or Disabled), and when the last event was sent (Last Event Time (UTC)).

To edit the subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Edit**.
- 3 If required, enter the new Workspace ID or secret key and click **Next**.
- 4 If required, add and remove the subsystems included in the subscription.
- 5 Click **Finish**.

To remove a subscription

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Select the required subscription and click **Delete**.
- 3 Confirm the removal.

To enable and disable a subscription

- When viewing the summary information, select the status column and choose to enable or disable the subscription as required.

To refresh the summary information

- 1 From the **Administration Tasks**, select **Configuration | Event Subscriptions**.
- 2 Click Refresh.

New-CA SentinelEventSubscription

Use this command to create the subscription required to send Change Auditor event data to Microsoft Sentinel.

Table 2. Available parameters

Parameter	Description
-WorkspaceID	The unique identifier for the Log Analytics workspace that has been enabled for Microsoft Sentinel.
-SecretKey	The primary or secondary key for the Log Analytics workspace that has been enabled for Microsoft Sentinel.
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems.

NOTE:

- If not specified, the default is to send events for all subsystems.
- To obtain an array of subsystems, use the [Get-CAEventExportSubsystems](#) command and filter the list to specify the required subsystems.

Parameter	Description
-StartTime (Optional)	Specifies date and time from which events should be sent. The default is to start sending events from the time when the subscription is created. For example: <ul style="list-style-type: none"> 20 July, 2020 12:01 PM uses local time 2020-07-20 12:10:00Z uses UTC time
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The default is 6500 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled. By default it is enabled.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to Microsoft Sentinel. By default this is set to 0 which results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the Heartbeat URL. Heartbeat notifications are sent to the Heartbeat_ChangeAuditor_CL log in Microsoft Sentinel where you can query and alert on them if required. By default, this is set to 0 (disabled).
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. By default, any coordinator can send the events. NOTE: The list order does not determine which coordinator is selected to send events.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include fields prefixed with additionalDetails, containing the values from the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included. By default, this is set to false.

Example: Create a subscription to send all subsystems event data to Microsoft Sentinel

```
New-CASentinelEventSubscription -Connection $connection -WorkspaceID $workspaceID -
SecretKey $secretKey -StartTime $dtStartTime -Subsystems $allowedSubsystems -
AllowedCoordinators $allowedCoordinators
```

Get-CASentinelEventSubscriptions

Use this command to see the details of the current Microsoft Sentinel subscriptions.

Table 22. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor PowerShell Command Guide for details.
-SubscriptionId (optional)	The ID of an existing Microsoft Sentinel subscription. If specified, the command will only return the Microsoft Sentinel subscription with that ID. If not specified, all Microsoft Sentinel subscriptions are returned.

Example: Get a list of all Microsoft Sentinel subscriptions

```
Get-CASentinelEventSubscriptions -Connection $connection
```

Command output

The command returns the following information.

Table 23. Available configuration information

Setting	Description
Id	The subscription ID.
WebhookSubscriptionId	The webhook subscription ID.
StartTimeUTC	Starting point in time for events being sent.
Subsystems	Subsystems that contain the event data being sent.
Enabled	Whether the subscription is enabled.
HeartbeatUrl	URL for heartbeat notifications.
LastEventTimeUTC	When the last event was sent.
LastEventResponse	Last event response.
LastHeartbeatTimeUTC	When the last heartbeat was sent.
LastHeartbeatResponse	The last heartbeat response. For example, statusCode = OK (200), statusCode = Bad Request (400), or statusCode = Internal Server Error (500).
EventsSent	Number of events sent.
BatchesSent	Number of batches sent.
HeartbeatsSent	Number of heartbeats sent.
NotificationInterval	How often how often (in milliseconds) notifications are sent.
HeartbeatInterval	How often (in milliseconds) heartbeat notifications are sent.
BatchSize	Batch size. (The maximum number of events that the active batch size can increase to.)
ActiveBatchSize	The current batch size. (The current number of events to include in a single notification message.) The batch size is automatically adjusted based on network throughput and system performance. Its value never exceeds the specified batch size.
AllowedCoordinators	List of coordinators permitted to send events or heartbeat notifications.
LastCoordinator	The last coordinator to send events or modify the subscription.
IncludeO365AADDDetails	Identifies whether or not the additionalDetails field with the raw JSON string is included for Office 365 and Azure Active Directory events. When set to true, the event will include fields prefixed with additionalDetails, containing the values from the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included.

Set-CASentinelEventSubscription

Use this command to modify a Microsoft Sentinel subscription

Table 2. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor PowerShell Command Guide for details.
-Subscription	The PSCASyslogEventSubscriptionStatus object that corresponds to the subscription to modify. This parameter is required if the SubscriptionId parameter is not specified.

Parameter	Description
-SubscriptionId	The ID of the subscription to modify. This parameter is required if the Subscription parameter is not specified. Use the Get-CASentinelEventSubscriptions command to find the ID.
-Subsystems (Optional)	Specifies an array of event subsystems from which to send events. This can be single or multiple subsystems. NOTE: <ul style="list-style-type: none"> To obtain an array of subsystems, use the Get-CAEventExportSubsystems command and filter the list to specify the required subsystems.
-WorkspaceId (Optional)	The unique identifier for the Log Analytics workspace that has been enabled for Microsoft Sentinel.
-SecretKey (Optional)	The primary or secondary key for the Log Analytics workspace that has been enabled for Microsoft Sentinel.
-BatchSize (Optional)	Specifies the maximum number of events to include in a single notification. The maximum is 6500 events.
-Enabled (Optional)	Specifies whether the subscription is enabled or disabled.
-NotificationInterval (Optional)	Specifies how often (in milliseconds) notifications are sent to Microsoft Sentinel. Setting to 0 results in a continuous stream of events.
-HeartbeatInterval (Optional)	Specifies how often (in milliseconds) heartbeat notifications are sent to the Heartbeat URL. Heartbeat notifications are sent to the Heartbeat_ChangeAuditor_CL log in Microsoft Sentinel where you can query and alert on them if required. Setting this to 0 disables the heartbeat message if enabled.
-AllowedCoordinators (Optional)	Specifies the DNS or NetBIOS name of the coordinators permitted to send events. If you specify a null or empty array, any coordinator can send the results. NOTE: The list order does not determine which coordinator is selected to send events.
-IncludeO365AADDetails (Optional)	Specifies whether to include the raw JSON event details provided by Microsoft. When set to true, the event will include fields prefixed with additionalDetails, containing the values from the raw JSON string for Office 365 and Azure Active Directory events. When set to false, the additionalDetails field is not included.

Example: Set a new batch size value for a Microsoft Sentinel subscription

```
Set-CASentinelEventSubscription -Connection $connection -SubscriptionId $subscriptionId -BatchSize $newIntValue
```

Remove-CASentinelEventSubscription

Use this command to remove a subscription.

Table 24. Available parameters

Parameter	Description
-Connection	A connection obtained by using the Connect-CAClient command. See the Change Auditor Command Guide for details.
-Subscription	The PSCASentinelEventSubscriptionStatus object obtained using Get-CASentinelEventSubscriptions that corresponds to the subscription to remove. This parameter is required if the SubscriptionId parameter is not specified.
-SubscriptionId	The ID of the subscription to remove. This parameter is required if the Subscription parameter is not specified. Use the Get-CASentinelEventSubscriptions command to find the ID.

Example: Remove a subscription

```
Remove-CASentinelEventSubscription -Subscription $subscription
```

Webhook technical insights

- [Handling webhook responses](#)

Handling webhook responses

To see the response codes, run the associated Get command and review the LastEventResponse and LastHeartbeatResponse in the output for the following response codes:

Table 1. Response codes

Response code	Description
HTTP 200	Notification successfully received This response code is expected for every notification.
HTTP 429	Too many events being sent When this occurs, Change Auditor will automatically reduce the batch size when it sends its next notification.
HTTP 400	Bad Request This occurs when the receiving server is unreachable or the data is improperly formatted. Review the information provided with the response for details.
HTTP 401	Unauthorized access For example, the notification message has an incorrect or expired AuthorizationID configured in the subscription. In this case, the subscription will be disabled until the error is corrected.
HTTP 500	Internal Server Error This can be either an issue with the Change Auditor coordinator or the receiving server.

Quest creates software solutions that make the benefits of new technology real in an increasingly complex IT landscape. From database and systems management, to Active Directory and Office 365 management, and cyber security resilience, Quest helps customers solve their next IT challenge now. Around the globe, more than 130,000 companies and 95% of the Fortune 500 count on Quest to deliver proactive management and monitoring for the next enterprise initiative, find the next solution for complex Microsoft challenges and stay ahead of the next threat. Quest Software. Where next meets now. For more information, visit www.quest.com.

Our brand, our vision. Together.

Our logo reflects our story: innovation, community and support. An important part of this story begins with the letter Q. It is a perfect circle, representing our commitment to technological precision and strength. The space in the Q itself symbolizes our need to add the missing piece—you—to the community, to the new Quest.

Contacting Quest

For sales or other inquiries, visit www.quest.com/contact.

Technical support resources

Technical support is available to Quest customers with a valid maintenance contract and customers who have trial versions. You can access the Quest Support Portal at <https://support.quest.com>.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request.
- View Knowledge Base articles.
- Sign up for product notifications.
- Download software and technical documentation.
- View how-to-videos.
- Engage in community discussions.
- Chat with support engineers online.
- View services to assist you with your product.